

Application/Control Number: 09/910,093
 Art Unit: 2654

Docket No.: 2001-0226

Amendments to the Claims:

This listing of claims will replace all prior versions, and listings, of claims in the application;

Listing of Claims:

1. (Currently Amended) A method of removing empty string terms from an automaton A having a set plurality of states "p", a set plurality of states "q", and a set plurality of outgoing transitions from the set plurality of states "p", E[p], the method comprising:

-inputting a plurality of electrical signals representing the automaton A, the automaton

A further representing a plurality of hypotheses with associated weights;

computing an ϵ -closure for each state of the plurality of states "p" of the automaton

A;

producing a plurality of electrical signals representing an improved automaton A, the

producing comprising modifying E[p] by:

removing each transition of the plurality of transitions labeled with an empty string; and

adding to the plurality of outgoing transitions, E[p], a non-empty-string transition, wherein each state of the plurality of states "q" is left with its weights pre-multiplied by an ϵ -distance from a corresponding one of the plurality of states state "p" to a respective one of the plurality of states state "q" in the automaton A.

2. (Currently Amended) The method of claim 1, wherein the producing of the plurality of electrical signals representing the improved automaton A further comprising comprises:

removing inaccessible ones of the plurality of the states "p" and "q" using a depth-first search of the automaton A.

BEST AVAILABLE COPY

Application/Control Number: 09/910,093
 Art Unit: 2654

Docket No.: 2001-0226

3. (Currently Amended) The method of claim 1, wherein adding to the plurality of outgoing transitions, $E[p]$, a non-empty-string ~~transitions~~ transition further comprises leaving each of the plurality of states "q" with weights $(d[p,q] \otimes p[q])$ to $E[p]$.

4. (Currently Amended) The method of claim 1, wherein ~~the step of~~ the computing of the ϵ -closure for each ~~input state of the plurality of states "p"~~ of an ~~the~~ input automaton A further comprises:

removing all transitions not labeled with an empty string from the automaton A to produce an automaton A_e ;

decomposing the automaton A_e into its strongly connected components; and

computing all-pairs shortest distances in each ~~component of the strongly connected~~ components visited in reverse topological order.

5. (Currently Amended) The method of claim 1, wherein ~~the step of~~ computing of the ϵ -closure for each ~~input state of the plurality of states "p"~~ of an ~~the~~ input automaton A further comprises:

removing all transitions not labeled with an empty string from the automaton A to produce an automaton A_e ;

decomposing A_e into its strongly connected components;

performing a single-source shortest-distance algorithm according to the following pseudo code:

```

for each  $p \in Q$ 
do  $d[p] \leftarrow r[p] \leftarrow \bar{0}$ 
 $d[s] \leftarrow r[s] \leftarrow \bar{1}$ 
 $S \leftarrow \{s\}$ 

```

Application/Control Number: 09/910,093
 Art Unit: 2654

Docket No.: 2001-0226

```

while S ≠ 0
do q ← head [S]
DEQUEUE (S)
r ← r[q]
r[q] ← 0̄
for each e ∈ E[q]
do if d[n[e]] ≠ d[n[e]] ⊕ (r ⊗ w[e])
then d[n[e]] ← d[n[e]] ⊕ (r ⊗ w[e])
r[n[e]] ← r[n[e]] ⊕ (r ⊗ w[e])
if n[e] ∉ S
then ENQUEUE (S, n[e])
d[s] ← T̄
    
```

6. (Currently Amended) The method of claim 1, wherein the ~~step of~~ computing of the ε -closure for each of the plurality of states ~~state~~ "p" further comprises computing each of the ~~ε -closure~~ ε -closures according to the following equation:

$$C[p] = \{(q, w) : q \in E[p], d[p, q] = w \in K - \{0\}\}.$$

7. (Currently Amended) The method of claim 6, ~~wherein the step of modifying outgoing transitions of each state "p"~~ further comprises modifying the outgoing transitions of each of the plurality of states ~~state~~ p according to the following procedure:

- (1) for each p ∈ Q
- (2) do E[p] ← {e ∈ E [p] : i[e] ≠ ε}
- (3) for each (q, w) ∈ C[p]
- (4) do E[p] ← E[p] ∪ {(p, a, w ⊗ w', r) : (q, a, w', r) ∈ E[q], a ≠ ε}

Application/Control Number: 09/910,093
 Art Unit: 2654

Docket No.: 2001-0226

- (5) if $q \in F$
 (6) then if $p \notin F$
 (7) then $F \leftarrow F \cup \{p\}$
 (8) $\rho[p] \leftarrow p[p] \oplus (\omega \oplus \rho[q])$.

8. (Currently Amended) The method of claim 7, wherein a state is a final state if ~~some~~ at least one of the plurality of states state "q" within a set of states reachable from one of the plurality of states "p" via a path labeled with an empty string is final and the final weight is

then:
$$\rho[p] = \bigoplus_{q \in \{p\}^* F} (d[p, q] \otimes \rho[q])$$

9. (Original) The method of claim 8, further comprising:
 performing a depth-first search of the automaton A after removing the empty strings.

10. (Currently Amended) A method of producing an equivalent weighted automaton "B" with no ϵ -transitions for any input weighted automaton "A" having at least one ϵ -transition, the automaton "A" having a set plurality of states "p", and a set plurality of states "q", the method comprising:

inputting a plurality of electrical signals representing the input weighted automaton A,
the weighted automaton A further representing a plurality of hypotheses with associated
weights;

computing an ϵ -closure for each state of the plurality of states "p" of the input weighted automaton "A"; and

producing a plurality of electrical signals representing the automaton "B" equivalent
to automaton A without the ϵ -transitions, the producing comprising:

modifying outgoing transitions of each state of the plurality of states "p" by:

Application/Control Number: 09/910,093
 Art Unit: 2654

Docket No.: 2001-0226

removing each transition labeled with an empty string; and
 adding to ~~each transition leaving state "p"~~ a plurality of outgoing transitions
from the plurality of states "p" a non-empty-string transition, wherein each state of the
plurality of states "q" is left with its weights pre-multiplied by an ϵ -distance from a
corresponding one of the plurality of states state "p" to a respective one of the
plurality of states "q" in the automaton "A" to produce the automaton "B"-equivalent
to automaton A without the ϵ -transitions, the automaton "B" representing an
improved version of the plurality of hypotheses.

11. (Currently Amended) The method of claim 10, further comprising:

removing inaccessible states of the automaton "A" using a depth-first search of the
 automaton "A".

12. (Currently Amended) The method of claim 11, wherein adding to the plurality of
outgoing transitions from the plurality of states "p" a non-empty-string transitions transition
 further comprises leaving each of the plurality of states state "q" with weights
 $(d[p, q] \otimes \rho[q])$ to the transitions leaving corresponding ones of the plurality of states "p".

13. (Currently Amended) ~~A~~ The method of claim 10, wherein the step of computing of
~~an~~ the ϵ -closure for each ~~input state~~ of the plurality of states "p" of ~~an~~ the input automaton
 "A" further comprises:

removing all non- ϵ -transitions to produce an automaton A_ϵ ;

decomposing the automaton A_ϵ into its strongly connected components; and

computing all-pairs shortest distances in each of the strongly connected components
~~component~~ visited in reverse topological order.

Application/Control Number: 09/910,093
 Art Unit: 2654

Docket No.: 2001-0226

14. (Currently Amended) The method of claim 10, wherein the ~~step of~~ computing of the ϵ -closure for each state of the plurality of states "p" further comprises computing each of the ϵ -closures according to the following equation:

$$C[p] = \{(q, w) : q \in \epsilon[p], d[p, q] = w \in K - \{\bar{O}\}\}.$$

15. (Currently Amended) The method of claim 14, wherein the ~~step of~~ modifying the outgoing transitions of each of the plurality of states state "p" further comprises modifying the outgoing transitions of each of the plurality of states state p according to the following procedure:

- (1) for each $p \in Q$
- (2) do $E[p] \leftarrow \{e \in E[p] : i[e] \neq \epsilon\}$
- (3) for each $(q, w) \in C[p]$
- (4) do $E[p] \leftarrow E[p] \cup \{(p, a, w \otimes w', r) : (q, a, w', r) \in E[q], a \neq \epsilon\}$
- (5) if $q \in F$
- (6) then if $p \notin F$
- (7) then $F \leftarrow F \cup \{p\}$
- (8) $\rho[p] \leftarrow \rho[p] \oplus (\omega \oplus \rho[q])$.

16. (Currently Amended) A method of producing an automaton B from an automaton A, the automaton B having no empty string transitions, the method comprising:

inputting a plurality of electrical signals representing the automaton A, the automaton A further representing a plurality of hypotheses with associated weights;

Application/Control Number: 09-910,093
 Art Unit: 2654

Docket No.: 2001-0226

computing for each state p in the automaton A its ϵ -closure $C[p]$ according to the following: $C[p] = \{(q, w) : q \in \epsilon[p], d[p, q] = w \in K - \{\emptyset\}\}$, where $\epsilon[p]$ represents states labeled with an empty string;

removing each transition labeled with an empty string; and

producing a plurality of electrical signals representing the automaton B, the automaton B being equivalent to the automaton A without ϵ -transitions, the producing comprising:

adding to each transition leaving the states state "p" a non-empty-string transition, wherein each state " q " in the automaton A is left with its weights pre-multiplied by an ϵ -distance from one of the states state "p" to a respective one of the states "q" in the automaton "A" to produce the automaton "B" equivalent to automaton A without the ϵ -transitions.

17. (Currently Amended) The method of claim 16, wherein the adding the non-empty-string transition to each of the transitions leaving the states "p" strings to $E[p]$ is performed according to the following code:

```
(+)   for each  $p \in Q$ 
(2)   do  $E[p] \leftarrow \{e \in E[p] : i[e] \neq \epsilon\}$ 
(3)   for each  $(q, w) \in C[p]$ 
(4)   do  $E[p] \leftarrow E[p] \cup \{(p, a, w \otimes w', r) : (q, a, w', r) \in E[q], a \neq \epsilon\}$ 
(5)   if  $q \in F$ 
(6)   then if  $p \in F$ 
(7)   then  $F \leftarrow F \cup \{p\}$ 
(8)    $\rho[p] \leftarrow \rho[p] \oplus (w \oplus \rho[q])$ , where  $E[p]$  is plurality of outgoing transitions from the states "p".
```

Application/Control Number: 09,910,093
 Art Unit: 2654

Docket No.: 2001-0226

18. (Currently Amended) The method of claim 10, further comprising modifying a plurality of outgoing transitions from the states "p", $E[p]$, according to the following procedure:

- (1) for each $p \in Q$
- (2) do $E[p] \leftarrow \{e \in E[p] : i[e] \neq \epsilon\}$
- (3) for each $(q, w) \in C[p]$
- (4) do $E[p] \leftarrow E[p] \cup \{(p, a, w \otimes w', r) : (q, a, w', r) \in E[q], a \neq \epsilon\}$
- (5) if $q \in F$
- (6) then if $p \notin F$
- (7) then $F \leftarrow F \cup \{p\}$
- (8) $\rho[p] \leftarrow \rho[p] \oplus (w \oplus \rho[q])$

19. (Currently Amended) A method of producing an equivalent weighted automaton "B" with no ϵ -transitions for any input weighted automaton "A" having a set of transitions E, wherein each transition "e" in the set of transitions has an input label $i[e]$, at least one transition being an ϵ -transition, a set of states P, each state in the set of states P is denoted as "p", and a set of states Q, each state in the set of states Q denoted as "q", a weight $w[e]$ for each transition "e", and $E[p]$ the transitions leaving each state "p" and $E[q]$ being the transitions leaving state "q", an ϵ -closure for a state being defined as $C[p]$, and where $\epsilon[p]$ represents a set of states reachable from state "p" via a path labeled with an ϵ -transition, the method comprising:

inputting a plurality of electrical signals representing the weighted automaton A, the weighted automaton A further representing a plurality of hypotheses with associated weights;

computing an ϵ -closure $C[p]$ for each state "p" of the input weighted automaton "A";

removing each ϵ -transition of the weighted automaton A to produce an automaton A_ϵ ;

and